

REMARKS

This responds to the Office Action dated February 2, 2006. Claim 25 is pending. Applicant has added new dependent claims 26-31, however, no new matter has been added. Individual issues raised by the Examiner are addressed next in the order in which they appear in the Office Action.

Information Disclosure Statement

In Paragraph 1, the Office Action alleged that the Information Disclosure Statement (IDS) filed on July 28, 2003 fails to comply with 37 CFR 1.98(a)(2), which requires a legible copy of each cited foreign patent document; each non-patent literature publication or that portion which caused it to be listed; and all other information or that portion which caused it to be listed. The Office Action further stated that the IDS has been placed in the application file, but the information referred to therein has not been considered.

Applicant respectfully requests that the information provided in the IDS filed on July 28, 2003 be fully considered because the IDS complies with the requirements of 37 CFR 1.98(d), as was indicated in Section 6 of the IDS, which is attached hereto as Exhibit A. 37 CFR 1.98(d) provides that:

(d) A copy of any patent, publication, pending U.S. application or other information, as specified in paragraph (a) of this section, listed in an information disclosure statement is required to be provided, even if the patent, publication, pending U.S. application or other information was previously submitted to, or cited by, the Office in an earlier application, unless:

(1) The earlier application is properly identified in the information disclosure statement and is relied on for an earlier effective filing date under 35 U.S.C. 120; and

(2) The information disclosure statement submitted in the earlier application complies with paragraphs (a) through (c) of this section.

In Section 6 of the IDS, (1) the earlier application was properly identified as U.S. Pat. App. No. 09/412,158, which is relied upon to establish an earlier effective filing date of the present application under 35 U.S.C. 120; and (2) the IDS submitted in the earlier application complied with paragraphs (a) through (c) of 37 CFR 1.98. Accordingly, applicant respectfully requests that the IDS submitted on July 23, 2003 be deemed compliant with 37 CFR 1.98 and all references listed therein be fully

considered. If the references submitted previously cannot be located for any reason, applicant will provide additional copies under a separate cover.

Rejection Under 35 U.S.C. 102

In Paragraphs 3-5 of the Office Action claim 25 was rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. patent 5,999,933 to Mehta (“Mehta”). Applicant respectfully disagrees.

Briefly, Mehta discloses a hardware/software system for analyzing memory dumps. (Mehta, *Abstract*). In the Mehta disclosure the user specifies a name of the software for which a memory dump was taken. (Mehta, col. 7, lines 46-53). Scanner software extracts all of the data structure definitions from symbol tables in the software, as known in the art. (Mehta, col. 7, lines 54-65). The extracted data structure definitions are stored in a text file called a template library (shown in Mehta, Fig. 5). Based on the user input and the contents of the template library, extraction software populates an extraction table (or logical table) in a database with information regarding the data structures in the dump. (Mehta, col. 8, lines 41-49). Among other information, the extraction table (shown in Mehta, Fig. 6) contains fetch logic parameters. (See Mehta, col. 8, lines 50-67). Although Mehta does not specifically describe what is meant by “fetch logic,” applicant believes that the term refers to a method of memory allocation at the time when a data structure is instantiated. In view of Mehta’s specification, it appears that by following a given fetch logic at the time of memory dump analysis, it is possible to determine the location (such as an address range) and values at those locations associated with the data structures in the memory dump.

Mehta discloses eleven types of fetch logic, with associated parameters. In some embodiments, the extraction software follows one of the fetch logics to retrieve portions of a memory dump associated with instances of data structures listed in the extraction table, and stores them in the database. (See Mehta, col. 20, lines 18-57; claim 9).

Mehta discloses different types of possible fetch logic associated with data structures in the extraction table. (See Mehta, col. 9, lines 1-54). A set of parameters, required to effectuate the logic is specified for every fetch logic. For example, using

fetch logic zero (called “Specify Address”) requires specifying parameters required to retrieve part of a memory dump at a particular address, such as result type, address entered by the user, etc. (See Mehta, col. 10, lines 1-53). Using fetch logic four (called “User Defined Function”), for example, requires specifying a function name and a macro name. (See Mehta, col. 10, lines 54-64).

Of particular interest to this discussion is fetch logic seven, which was cited in the Office Action. The pertinent disclosure of fetch logic seven in Mehta is listed below:

An additional option in Table IX is fetch logic seven. Fetch logic seven is for determining the number of instances of data structures of a selected type from the number of instances of a reference data structure. In fetch logic seven, the number of instances is equal to the number of instances of another structure. Parameter Text1 is the template library name that contains the definition of the reference data structure. Parameter Text2 is the name of the reference data structure in the template library of parameter Text1. Parameter Int1 is the integer offset to parameter Text2 within the template library 152' designated by parameter Text1. The part in the template library 152' at offset Int1 describes a field in the reference data structure that contains the number of occurrences that this reference data structure has. (Mehta, col. 14, lines 4 - 17).

TABLE IX		
Method of Obtaining Numbers Of Instances Of Data Structure Type (1318 in FIG. 13)		
Fetch Logic	Parameters	Description
0	see TABLE II	(1322)
1	see TABLE IV	(1330)
2	see TABLE V	(1334)
7	Text1	Template Library Name (not shown)
7	Text2	Data Structure Type Name (not shown)
7	Int1	Offset Within Template Library (not shown)
8	Int1	Header Offset (1342)
9	Int1	Absolute Value (1346)

Based on the above, applicant respectfully submits that Mehta does not anticipate claim 25. First, Mehta does not disclose “the database having been compressed by storing information regarding distinct values of an attribute and information regarding the number of occurrences of distinct values.” As noted, Mehta discloses (1) a standard database management system (with a database), such as MS Access for storing extraction tables and (2) “template library, which is a text file,” used for creation of extraction tables. (See e.g., Mehta, Fig. 1 showing template libraries to be separate from extraction tables and from the database management system and a

database.) There is no disclosure in Mehta that either of these two distinct items is a compressed database.

Second, steps (a)-(d) of claim 25 are related to the structure of the compressed database recited in the claim preamble. By contrast, Mehta relates neither to the structure nor to any compression of a database. In particular, claim 25 recites that “the database having been compressed by storing (1) information regarding distinct values of an attribute and (2) information regarding the number of occurrences of distinct values . . .” Mehta does not disclose “retrieving information regarding the number of occurrences of a given element,” recited in step (a) of claim 25, at least because Mehta does not disclose compressing a database by storing information regarding the number of occurrences. As recited in claim 25, the information regarding the number of occurrences of a given value retrieved in step (a) is stored in the database (accomplishing compression), which, as mentioned above, Mehta simply does not disclose.

Similarly, step (b) of claim 25 recites “determining an instance element based on information regarding the number of occurrences of the given value,” retrieved in step (a). The information determined in step (b) of claim 25 is linked to the information stored in the database, also related to database compression. Mehta provides no disclosure corresponding to step (b). The Office Action’s reliance on Mehta for disclosing “determining an instance element” is incorrect, at least because the cited portion of Mehta’s specification at col. 14, lines 4-17 discloses “determining the number of instances,” but not “an instance element,” as recited in the claim.

Mehta also fails to disclose steps (c) – (d) recited in claim 25, because determining a connectivity element in step (c) is based on the instance element determined in step (b). As discussed, this instance element is based on information regarding the number of occurrences of a given value, which is tied to the structure of the compressed database. As to step (d), the determined record is based on the connectivity element, which, as shown above, is also based on the information stored in the compressed database. Mehta provides no such disclosures.

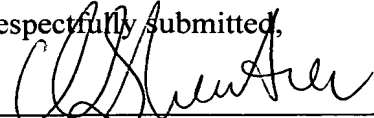
Accordingly, for the forgoing reasons Mehta cannot anticipate independent claim 25. Applicant respectfully requests that the corresponding rejection be withdrawn.

Conclusion

In light of the above remarks, the applicant respectfully requests that the Examiner reconsider this application with a view towards allowance. The Examiner is invited to call the undersigned attorney, if a telephone call could help resolve any remaining items.

Date: August 2, 2006

Respectfully submitted,



Ognjan V. Shentov

38,051
(Reg. No.)

JONES DAY

222 East 41st Street

New York, New York 10017

(212) 326-3939